

AMES GRANT / IN-33

76038

COMPUTER SYSTEMS LABORATORY

STANFORD UNIVERSITY · STANFORD, CA 94305-2192



p. 45

THROUGHPUT ANALYSIS OF THE IEEE 802.4 TOKEN BUS STANDARD UNDER HEAVY LOAD

Joseph Pang and Fouad Tobagi

Technical Report: CSL-TR-87-331

March 1987

(NASA-CR-180936) THROUGHPUT ANALYSIS OF THE
IEEE 802.4 TOKEN BUS STANDARD UNDER HEAVY
LOAD (Stanford Univ.) 45 p Avail: NTIS
PC AC3/MF AC1 CSCL 09C

N87-22921

Unclas
G3/33 0076038

This work was supported by the National Aeronautics and Space Administration
under Grants No. NAG2-292 and No. NAGW-419.

THROUGHPUT ANALYSIS OF THE IEEE 802.4 TOKEN BUS STANDARD UNDER HEAVY LOAD

Joseph Pang and Fouad Tobagi

Technical Report: CSL-TR-87-331

May 1987

**Computer Systems Laboratory
Departments of Electrical Engineering and Computer Science
Stanford University
Stanford, California 94305-2191**

Abstract

It has become clear in the last few years that there is a trend towards integrated digital services. Parallel to the development of public Integrated Services Digital Network (ISDN) is service integration in the local area (*e.g.* a campus, a building, an aircraft). The types of services to be integrated depend very much on the specific local environment. However, applications tend to generate data traffic belonging to one of two classes. According to IEEE 802.4 terminology, the first major class of traffic is termed synchronous, such as packetized voice and data generated from other applications with real-time constraints, and the second class is called asynchronous which includes most computer data traffic such as file transfer or facsimile.

In this report, we examine the IEEE 802.4 token bus protocol which has been designed to support both synchronous and asynchronous traffic. The protocol is basically a timer-controlled token bus access scheme. By a suitable choice of the design parameters, it can be shown that access delay is bounded for synchronous traffic. As well, the bandwidth allocated to asynchronous traffic can be controlled. We present a throughput analysis of the protocol under heavy load with constant channel occupation of synchronous traffic and constant token-passing times.

1. Introduction.

The phenomenal development of sophisticated digital technology has made major impacts on the communication industry. Rapid transitions to digital communication services are being made in local and long-haul public networks. The concept of using the same network to support multiple services such as voice, video, computer data, *etc.*, has become more attractive than ever before. In the public domain, standards are being drawn for ISDN. At the same time, alternatives for service integration in the local area are being studied. One possible approach is to use advanced circuit switching techniques. Another approach is to employ a local area network (LAN). We are interested in the latter approach in this report.

Traditionally, LAN's have been designed to handle bursty computer traffic through statistical multiplexing techniques. The incorporation of real-time traffic introduces stringent constraints on delay. For example, packetized voice has a deadline requirement. Voice packets that cannot be delivered within a certain deadline are discarded, leading to degradation in voice quality. In an automated manufacturing environment, data packets for machine control purposes must be delivered in time to avoid undesirable or sometimes disastrous effects. In a distributed computing environment, the performance of the system may very much rely on tight synchronization of processes which communicate with each other through a local network. The strict delay constraints required by these applications cannot be adequately met by LAN protocols originally designed for conventional computer communication.

There have been various studies and proposals of LAN protocols designed to support real-time applications together with conventional computer applications [7]–[18]. In this report, we examine one of these proposals, namely, the IEEE 802.4 token bus standard [18]. In this standard, timers are introduced to control channel access in addition to the basic token bus multi-access protocol. As a result, the protocol has the capability of accommo-

dating both synchronous (time-critical) and asynchronous (non-time-critical) traffic. The idea behind this protocol is similar to other cycle-limiting token-passing schemes [12]–[15]. The basic design parameters in the IEEE 802.4 protocol are called token hold times and target token rotation times. We shall investigate the effects of these system parameters on the throughput performance of different classes of traffic.

In the next section, we shall give a brief description of the IEEE 802.4 token bus protocol under normal fault-free conditions, paying particular attention to the timer control mechanism. In section 3, we develop a generalized model to study the characteristics of this protocol. In section 4, we present the major analytic results that reveal how different users share the channel bandwidth under the heavy load situation. By a suitable choice of the design parameters, synchronous traffic is guaranteed a bounded access delay and a fixed service duration for each channel access. Furthermore, asynchronous traffic shares the excess bandwidth according to the target token rotation times assigned. Finally, we conclude with a summary.

2. Description of the IEEE 802.4 Protocol under Normal Operation Conditions.

The access protocol described in the IEEE 802.4 standard [18] is based on token-passing on a bus. Under fault-free conditions, a logical ring is maintained and the token is passed according to the logical ring. Considerable overhead is needed to maintain the logical ring against communication errors and station failures. Since we are only interested in the characteristics of the protocol under normal conditions, we shall assume a logical ring has been established and no error conditions are present. The interested readers should refer to [18] for a thorough description of the error recovery functions. Of particular interest in this protocol is the time-out mechanism that limits the channel access of each asynchronous class.

In the IEEE 802.4 token bus standard, there are four globally defined classes of services, labelled 0, 2, 4 and 6. Each station in the logical ring is given full access to all four classes of services. Class 6 service is provided for the so-called synchronous traffic (*e.g.* voice) and other classes are for asynchronous traffic (*e.g.* file transfer). For convenience, we shall classify class 6 service or traffic as type I and other classes as type II. A time value called token hold time (*THT*) is assigned for class 6 service and 3 other time values called target token rotation times (*TRT*'s) are assigned for classes 0, 2 and 4.

Each station is equipped with 4 loadable counters, C_0, C_2, C_4, C_6 . Each counter counts down continuously from a positive value to zero. When a counter reaches zero, it remains at zero until a new value is loaded. Counters C_0, C_2, C_4 are associated with classes 0, 2, 4 respectively. Upon reception of a token, a station loads C_6 with *THT* and transmits class 6 packets until C_6 expires or no further class 6 packets remain. If C_6 expires during a packet transmission, the current transmission will nevertheless be completed. After servicing class 6 traffic, the value of C_4 will be loaded into C_6 and immediately the *TRT* assigned to class 4 will be loaded into C_4 . Then class 4 is serviced until C_6 expires or no

further class 4 traffic. The above procedure is repeated for class 2 and then for class 0. Then the station releases the token to the next logical station.

As we can see, the use of timers to control service requires minimal modification of the standard token-passing protocol. As well, it does not require any station to broadcast their status as in many reservation schemes. Thus, the *THT/TRT* scheme is most attractive both in terms of cost and robustness. The question is how well can the *THT/TRT* scheme support multiple types of traffic.

Although the detailed behaviour of the above *THT/TRT* scheme is not obvious, the philosophy is clear. The time-out mechanism ensures a finite bound on the cycle, that is, the period between successive token receptions at a specific station. Upon reception of the token, each station is guaranteed class 6 service of duration equal to *THT*. Packets of any other class can gain access to the channel only if the associated counter of that class does not expire before loading it into C_6 . This means that if a given cycle is unusually large, then services of classes 0, 2 and 4 will be deferred which would tend to reduce the next cycle.

Clearly, class 6 at a station has a somewhat preferred status over other classes since it is guaranteed non-zero service in each cycle. The differences among the *TRT*'s of classes 0, 2 and 4 restrict the use of the channel by these classes to different extents. We shall derive analytic results regarding the throughput of each class under heavy load.

3. Queueing Model.

3.1 General Descriptions of Model.

The first step to construct a versatile analytic model for the IEEE 802.4 access scheme is to consider each class at a station to be a separate queue in the system. Thus, we do not have the concept of a station any more. When the token arrives at a queue, the queue will be serviced according to the *THT/TRT* protocol. With this abstraction, we are led to consider the general model of N queues serviced by a single server in a cyclic order, that is, 1 2 3 ... N , then back to 1 and repeat the pattern. In this general model, N is an arbitrary positive integer. Furthermore, different type I queues may have different *THT*'s. Similarly, type II queues may be assigned different *TRT*'s.

In token-passing with the cyclic order described above, the k th *cycle* of queue i is the period between the k th and $(k + 1)$ st receptions of the token by queue i . Thus, the cycles associated with different queues are not the same. According to the *THT/TRT* protocol, in each cycle a type I queue can receive service up to its *THT* or until the queue is empty. However, a type II queue can only receive service up to its *TRT* less the length of the previous cycle or until the queue becomes empty; if the length of the previous cycle exceeds *TRT*, then that queue will not receive any service in the current cycle. If the service timer expires during a packet transmission, the transmission shall nevertheless be completed. This means that the actual service time of a queue can exceed its quota; we shall call this the *overflow effect*. After servicing a queue, the server moves to the next logical queue; the time it takes the server to move from a given queue to the next is called the *walk time* or *switching overhead*.

We limit our study of the above system when all type II queues are heavily loaded, that is, every type II queue always has packets to send. We further assume the service of

a type I queue is constant over all cycles and that the walk time from a given queue to the next is constant over all cycles as well. The overflow effect is assumed to be negligible, that is, a queue cannot receive service exceeding its quota in any given cycle. Finally, we consider a time-slotted system where all time quantities are measured in integer multiple of slots. This is not a major limitation since the size of the basic time slot can be arbitrarily small to accommodate continuous systems.

3.2 System Variables and Parameters.

In this sub-section, we define the following non-negative integer-valued system variables and parameters. Let

THT_i = THT assigned to queue i if it is of type I,

TRT_i = TRT assigned to queue i if it is of type II,

W_i = Switching overhead or walk time from queue i to its logical successor (assumed independent of k),

H_i = Service time of queue i if it is of type I (assumed independent of k),

$C_i^{(k)}$ = Length of the k th cycle of queue i ,

$T_i^{(k)}$ = Service time of queue i in the k th cycle.

Using the above definitions, we can write down the following relations with summations defined as zero if the lower summation index exceeds the upper index. The length of the k th cycle of queue i is given by

$$C_i^{(k)} = \sum_{j=i}^N (T_j^{(k)} + W_j) + \sum_{j=1}^{i-1} (T_j^{(k+1)} + W_j) \quad (3.1)$$

For type I queues,

$$T_i^{(k+1)} = H_i \quad (3.2a)$$

and for type II queues,

$$T_i^{(k+1)} = \max(TRT_i - C_i^{(k)}, 0) \quad (3.2b)$$

3.3 Imbedded Markov Chain.

The recursive relations (3.1), (3.2a) and (3.2b) among service times and cycle times show that the vector $\vec{T}^{(k)} = (T_1^{(k)}, T_2^{(k)}, \dots, T_N^{(k)})$ constitutes a deterministic Markov Chain. Given $\vec{T}^{(k)}$, we can compute $C_1^{(k)}$ from (3.1) and then $T_1^{(k+1)}$ from (3.2a) or (3.2b). Similarly, $T_2^{(k+1)}, \dots, T_N^{(k+1)}$ can be successively calculated using the same relations. Thus, we can obtain $\vec{T}^{(k+1)}$ from $\vec{T}^{(k)}$ deterministically.

Since the service times H_i of type I queues are constant over all cycles, they appear as part of switching overheads as far as type II queues are concerned. Thus, we can consider the equivalent system consisting only of type II queues with modified switching overheads without loss of generality.

We can further transform the system to one that has zero overheads by rewriting (3.1) and (3.2b) as

$$C_i^{(k)} - W = \sum_{j=i}^N T_j^{(k)} + \sum_{j=1}^{i-1} T_j^{(k+1)} \quad (3.3)$$

$$T_i^{(k+1)} = \max((TRT_i - W) - (C_i^{(k)} - W), 0) \quad (3.4)$$

where

$$W = \sum_{i=1}^N W_i \quad (3.5)$$

We disregard equation (3.2a) since we are considering the equivalent system with only type II queues and modified overheads. Relations (3.3) and (3.4) correspond to a system with no switching overheads, TRT_i modified as $(TRT_i - W)$ and $C_i^{(k)}$ modified as $(C_i^{(k)} - W)$.

If the modified TRT_i is less than zero for queue i , then queue i will not receive any service at all. Such queues can be effectively removed from the system. Therefore, it is sufficient to consider the system with only type II queues with positive modified TRT and zero switching overheads. We shall devote most of our attention to this equivalent system. Nevertheless, we should bear in mind the relations between the original system and its equivalent so that we can convert one from another without any difficulty.

3.4 Definitions and Terminology.

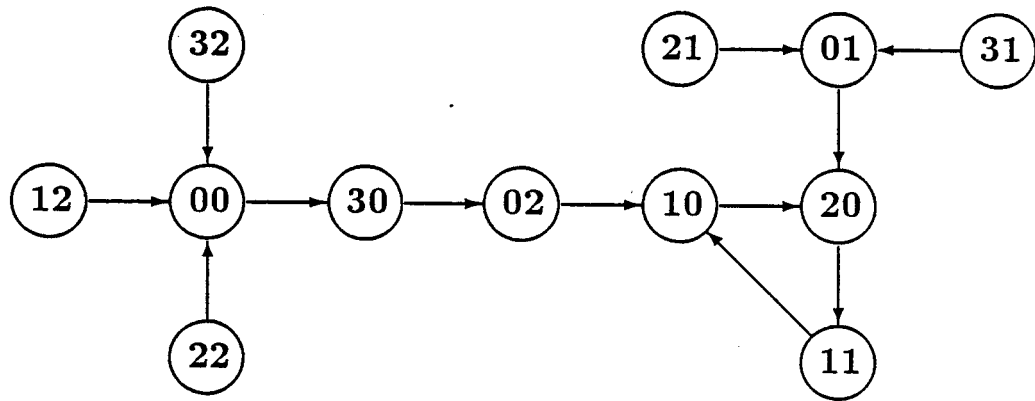
With the discussion from last sub-section, we restrict ourselves to systems with only type II queues with positive TRT and no switching overheads. The Markov relations (3.1) and (3.2b) can now be written as

$$C_i^{(k)} = \sum_{j=i}^N T_j^{(k)} + \sum_{j=1}^{i-1} T_j^{(k+1)} \quad (3.6)$$

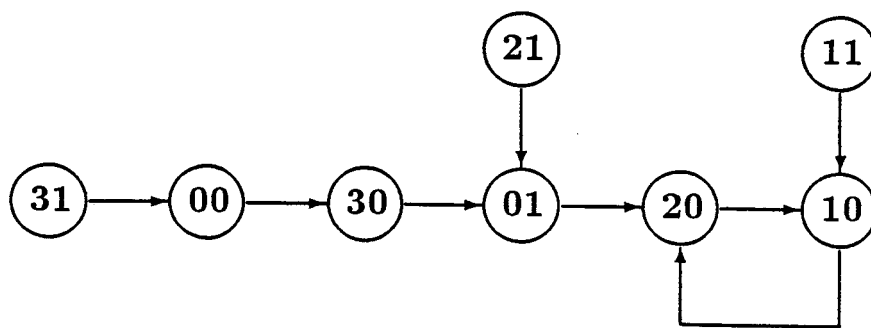
$$T_i^{(k+1)} = \max(TRT_i - C_i^{(k)}, 0) \quad (3.7)$$

Our goal is to study the transitions of the state vector $\vec{T}^{(k)}$. The Markov relations (3.6) and (3.7) can be equivalently expressed by a state-transition diagram. Since $T_i^{(k)} \leq TRT_i$ for all k and i , the set of possible state vectors is finite. It is therefore possible to construct the state-transition diagram consisting of all state vectors. Let us consider a few examples.

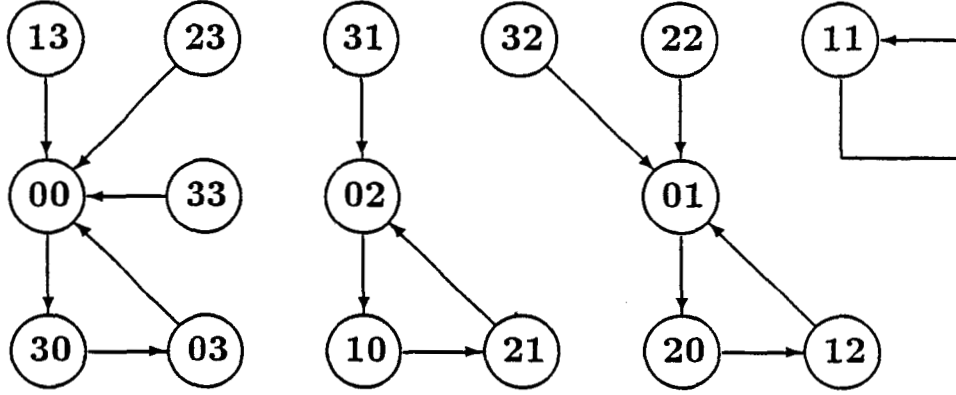
Example 3.1 $N = 2$, $TRT_1 = 3$, $TRT_2 = 2$.



Example 3.2 $N = 2$, $TRT_1 = 3$, $TRT_2 = 1$.



Example 3.3 $N = 2$, $TRT_1 = TRT_2 = 3$.



Clearly, the number of possible state vectors can be extremely large even for moderate values of N and TRT_i . The challenge is to find a simple characterization of the unwieldy state-transition diagram.

We define a set of terms which will be used later, but we advise the reader not to linger too long on the significance of these definitions in the first reading. As we progress in the next section, the reader shall see the necessity of defining these terms; at that point, the reader can refer to the definitions in this sub-section.

Classification of states.

We call the N -tuple $(T_1^{(k)}, T_2^{(k)}, \dots, T_N^{(k)})$ of service times a *state* in the k th cycle. The $(N+1)$ -tuple $(T_1^{(k)}, T_2^{(k)}, \dots, T_N^{(k)}, T_1^{(k+1)})$ is the *extended state* in the k th cycle. State s_2 is *accessible* or *reachable* from state s_1 if starting from s_1 , it is possible to enter s_2 after a finite number of state transitions. States that are reachable from at least one state are *reachable states*, otherwise they are *unreachable*. States are *communicating* if they are

reachable from one another. Communicating states are also known as *recurrent states*. States that are not recurrent are *transient states*. A set of communicating states is called a *recurrent set* or a *loop*.

Classification of queues.

Next, we classify the queues according to their *TRT*'s. Let $TRT_i \in \{\alpha_1, \alpha_2, \dots, \alpha_M\}$ for all i where $\alpha_1 > \alpha_2 > \dots > \alpha_M > 0$. A queue with *TRT* equal to α_k is said to have priority k . A k -priority queue is said to have a higher priority than a $(k+i)$ -priority queue for positive i . We let N_k denote the number of k -priority queues in the system. It is also convenient to let the indices of 1-priority queues to be p_i for $1 \leq i \leq N_1$ where p_i is an increasing sequence. Queue p_i is called the i th 1-priority queue.

System operating modes.

With the above terminology, we can classify recurrent sets into *depletion* and *non-depletion* types. A *depletion recurrent set* is a loop in which no queues, except the ones with priority 1, receive any service. In such a case, the system is said to operate in *depletion mode*, otherwise the system is operating in *non-depletion mode*. The loop in example 3.2 is of the depletion type whereas the loop in example 3.1 is of the non-depletion type. In a depletion loop, the 1-priority queues are depleting the services of other queues.

Pre-group and post-group.

We now introduce the ideas of *pre-group* and *post-group* for the case when queue 1 has priority 1. Let $p_1 = 1$. The extended state vector $(\vec{T}^{(k)}, T_1^{(k+1)})$ can be partitioned into $(N_1 + 1)$ sub-vectors according to the positions of the 1-priority queues. There are two conceivable ways of performing the partition. The first way is such that the first component of each sub-vector corresponds to the service of a 1-priority queue; these sub-vectors are called *pre-groups*. The second way is such that the last component of each sub-vector corresponds to the service of a 1-priority queue; these sub-vectors are called *post-groups*.

Let $U_1^{(k)}, U_2^{(k)}, U_3^{(k)}, \dots, U_{N_1}^{(k)}, U_{N_1+1}^{(k)}$ be a partition of the extended state vector $(\vec{T}^{(k)}, T_1^{(k+1)})$, that is, $(\vec{T}^{(k)}, T_1^{(k+1)}) = (U_1^{(k)}, U_2^{(k)}, \dots, U_{N_1}^{(k)}, U_{N_1+1}^{(k)})$. If the first component of each $U_i^{(k)}$ corresponds to the service of a 1-priority queue, then $U_i^{(k)}$ is the i th *pre-group* of the k th cycle for $1 \leq i \leq N_1$ and $U_{N_1+1}^{(k)}$ is the 0th *pre-group* of the k th cycle. The first component of each pre-group is called the *group leader*. Similarly, if the last component of each $U_i^{(k)}$ corresponds to the service of a 1-priority queue, then $U_i^{(k)}$ is called the $(i-1)$ th *post-group* of the k th cycle for $1 \leq i \leq N_1 + 1$. The last component of each post-group is called the *group trailer*.

As an example, let $N = 10$ and queue 1, 3, 4, 9 be 1-priority queues while others are of lower priorities. Then $(T_1^{(k)}, T_2^{(k)})$, $(T_3^{(k)})$, $(T_4^{(k)}, T_5^{(k)}, \dots, T_8^{(k)})$, $(T_9^{(k)}, T_{10}^{(k)})$ and $(T_1^{(k+1)})$ are the 1st, 2nd, 3rd, 4th and the 0th pre-groups in the k th cycle, and that $(T_2^{(k+1)}, T_3^{(k+1)})$, $(T_4^{(k+1)})$, $(T_5^{(k+1)}, T_6^{(k+1)}, \dots, T_9^{(k+1)})$, $(T_{10}^{(k+1)}, T_1^{(k+2)})$ and $(T_1^{(k+1)})$ are the 1st, 2nd, 3rd, 4th and 0th post-groups in the $(k+1)$ st cycle.

Surplus and deficit.

Finally, we would like to introduce the concepts of *surplus* and *deficit* for a system with at least one 2-priority queue. Let queue i be a 1-priority queue. The component $T_i^{(k)}$ of the state vector $\vec{T}^{(k)}$ is said to have a zero deficit and a surplus of $(\alpha_1 - \alpha_2) - T_i^{(k)}$ if $T_i^{(k)} \leq (\alpha_1 - \alpha_2)$, otherwise it has zero surplus and a deficit of $T_i^{(k)} - (\alpha_1 - \alpha_2)$. For queue j of a lower priority, the deficit of the component $T_j^{(k)}$ is given by $T_j^{(k)}$ and the surplus of $T_j^{(k)}$ is always zero. By definition, surplus and deficit are both non-negative quantities.

The surplus of a state is the sum of surpluses of all its components and the deficit of a state is the sum of deficits. The surplus and deficit for a pre-group, a post-group and an extended state are defined analogously.

4. Analytic results.

In the previous section, we have formulated step-by-step a state transition problem. We defined a state and an extended state, a pre-group and a post-group, depletion loops and non-depletion loops. We also introduced the concepts of surplus and deficit. In this section, we shall make use of these concepts to derive results concerning the bandwidth allocated to each queue under heavy load.

4.1 Bound on Cycle Length.

The purpose of the TRT 's is to limit service times of type II queues to guarantee a maximum access delay of type I queues. This is demonstrated by the following proposition.

Proposition 4.1 The sum of components of a reachable state is less than or equal to α_1 , the largest TRT assigned.

Proof

Let $\vec{T}^{(k+1)}$ be an arbitrary reachable state and $\vec{T}^{(k)}$ be the previous state. From equations (3.6) and (3.7), we have

$$T_n^{(k+1)} = \max\left(TRT_n - \sum_{l=n}^N T_l^{(k)} - \sum_{l=1}^{n-1} T_l^{(k+1)}, \quad 0\right) \quad (4.1)$$

Adding $\sum_{l=1}^{n-1} T_l^{(k+1)}$ to both sides, we get

$$\sum_{l=1}^n T_l^{(k+1)} = \max\left(TRT_n - \sum_{l=n}^N T_l^{(k)}, \quad \sum_{l=1}^{n-1} T_l^{(k+1)}\right) \quad (4.2)$$

The above relation is an iterative one in the partial sums of $T_l^{(k+1)}$. By successive substitutions, we have

$$\sum_{l=1}^N T_l^{(k+1)} = \max\left(\max_{1 \leq m \leq N} \left(TRT_m - \sum_{l=m}^N T_l^{(k)}\right), \quad 0\right) \quad (4.3)$$

Since $T_l^{(k)} \geq 0$ for $1 \leq l \leq N$, we have

$$\sum_{l=1}^N T_l^{(k+1)} \leq \max_{1 \leq m \leq N} TRT_m = \alpha_1 \quad (4.4)$$

□

The above result shows that the cycle time of queue 1 is less than or equal to α_1 after the system has run for at least one cycle. Clearly, the result also applies to other queues by cyclic shifts of queue indices, that is, $C_l^{(k)} \leq \alpha_1$ for $1 \leq l \leq N$ and $k > 1$. In fact, the result holds true even when the type II queues are not heavily loaded. The proof is exactly the same except the equalities (4.1)–(4.3) are replaced by inequalities.

We can translate Proposition 4.1 from the equivalent system to the original system. Let $\alpha_1^{(org)}$ be the largest TRT in the original system, W be the total walk time and H be the total type I service in a cycle. If $\alpha_1^{(org)} < (H + W)$, then no type II queues will receive any service and all cycle times are equal to $(H + W)$. If $\alpha_1^{(org)} > (H + W)$, then the largest TRT in the equivalent system is given by $\alpha_1^{(eq)} = \alpha_1^{(org)} - (H + W)$. Since the cycle times in the equivalent system is no larger than $\alpha_1^{(eq)}$, the cycle times in the original system must be no larger than $\alpha_1^{(eq)} + (H + W)$ which is equal to $\alpha_1^{(org)}$. Therefore we conclude that the cycle times of all queues, type I or type II, in the original system are less than or equal to $\max(\alpha_1^{(org)}, H + W)$.

The access delay of a packet is defined as the time elapsed after the packet has reached the head of the queue until it receives service. Clearly, the access delay is less than or equal to the cycle time for type I queues; if the cycle time is bounded, then the access delay is also bounded. This is indeed the desirable feature common to all cycle-limiting token-passing schemes. It is noted that type II queues may not have bounded access delay even though the cycle times are bounded.

4.2 Symmetric Type II Queues.

By symmetry, if all queues are assigned the same TRT , then there should not be any difference in the services of each queue. This is indeed the case as shown by the following result.

Proposition 4.2 If $TRT_i = \alpha_1$ for all i , then all queues will have the same throughput.

Proof

Let us compute $\vec{T}^{(k+1)} = (j_1, j_2, \dots, j_N)$ from $\vec{T}^{(k)} = (i_1, i_2, \dots, i_N)$ where $k > 1$. Since $\vec{T}^{(k)}$ is a reachable state, we have $\sum_{k=1}^N i_k \leq \alpha_1$. Thus, $j_1 = \alpha_1 - \sum_{k=1}^N i_k$. Successive calculations of the other components show that $j_2 = i_1, j_3 = i_2, \dots, j_N = i_{N-1}$. We can therefore construct subsequent states in a similar fashion and display them in matrix form

$$\begin{array}{ccccccc}
 i_1 & i_2 & i_3 & i_4 & \cdots & i_{N-1} & i_N \\
 j_1 & i_1 & i_2 & i_3 & \cdots & i_{N-2} & i_{N-1} \\
 i_N & j_1 & i_1 & i_2 & \cdots & i_{N-3} & i_{N-2} \\
 i_{N-1} & i_N & j_1 & i_1 & \cdots & i_{N-4} & i_{N-3} \\
 \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
 i_4 & i_5 & i_6 & i_7 & \cdots & i_1 & i_2 \\
 i_3 & i_4 & i_5 & i_6 & \cdots & j_1 & i_1 \\
 i_2 & i_3 & i_4 & i_5 & \cdots & i_N & j_1
 \end{array}$$

The rows of the above matrix represent consecutive state vectors. Clearly, the matrix recurs as we continue the listing. Thus, the matrix represents a loop of the system. Since $\vec{T}^{(k)}$ is an arbitrary reachable state, we conclude that all loops of the system must have the structure of the above matrix. Furthermore, all reachable states are recurrent states.

The average service for the l th queue per cycle is given by the sum of elements of the l th column divided by $(N + 1)$ which is $\alpha_1/(N + 1)$. The average cycle length of any queue is the sum of all the elements divided by $(N + 1)$ which is $(N\alpha_1)/(N + 1)$. The throughput is given by the ratio of the average service per cycle to the average cycle length. Thus, the throughput of each queue in this symmetric equivalent system is $1/N$.

□

Let us look into the detailed structure of the matrix display of a loop. The matrix is an $(N + 1)$ by N matrix with at most $(N + 1)$ different elements. If we list the elements in a linear fashion by concatenating successive rows, we shall see the extended state $(i_1, i_2, \dots, i_N, j_1)$ being replicated N times. As well, each column of the above matrix can be obtained by cyclic shifts of the extended state. The successive cycle lengths of each queue are, in cyclic order, $(\alpha_1 - j_1), (\alpha_1 - i_N), (\alpha_1 - i_{N-1}), (\alpha_1 - i_{N-2}), \dots, (\alpha_1 - i_1)$.

If we define the period of a vector p to be the smallest number such that cyclic shifting the vector p times will yield the same vector, then it is clear that the period of a loop is the period of its extended state vectors. The largest possible period of a loop is $(N + 1)$. Shorter periods are possible but they must be divisors of $(N + 1)$. One can construct all possible loops by examining all $(N + 1)$ -tuples $(i_1, i_2, \dots, i_N, j_1)$ such that the sum of components is α_1 ; tuples that are cyclic shifts of each other belong to the same loop.

Notice that the throughput of a type II queue in the original system is less than $1/N_{II}$ where N_{II} is the number of type II queues in the original system. Using the definitions in sub-section 4.1, the average service of type II queues per cycle is

$$\bar{S}^{(org)} = \max\left(\frac{\alpha_1^{(org)} - (H + W)}{N_{II} + 1}, 0\right) \quad (4.5)$$

and the average cycle time in the original system is given by

$$\bar{C}^{(org)} = (H + W) + N_{II}\bar{S}^{(org)} \quad (4.6)$$

From the above equations, it is clear that the throughput of a type II queue is less than $1/N_{II}$. Furthermore, the throughput of a type II queue decreases as H or W increases, but increases as $\alpha_1^{(org)}$ increases. Thus, we see the tradeoff among the number of type I queues the system can support, the required bound on access delay and the throughput of type II queues.

4.3 Existence of Depletion Loops.

In the last sub-section, we have found a simple characterization of all loops for a symmetric system. Next we would like to consider the general asymmetric system, that is, one with at least one 2-priority queue. Our goal is again to study the loops that determine the behaviour of the system after the system has run for an arbitrarily long period of time.

The state space of a general asymmetric system, however large, is finite. Thus, there is at least one loop in the state-transition diagram. Our first step to study the loops of a general asymmetric system is to find out whether it is possible for the 1-priority queues to deplete services of all other queues, that is, whether there exist depletion loops.

Recall from sub-section 3.4 that in a depletion loop all queues other than 1-priority queues receive zero services. Thus the matrix display of a depletion loop, if there is one, will be similar to the matrix display of a loop of a symmetric system except we have to insert missing columns of zeros for the low priority queues. With this picture in mind, we can deduce a necessary and sufficient condition for the existence of depletion loops.

Proposition 4.3 For a system with at least one 2-priority queue, depletion loops exist if and only if $(N_1 + 1)(\alpha_1 - \alpha_2) \geq \alpha_1$.

Proof

If there exists a depletion loop, let i_l be the service of the l th 1-priority queues in a state within this loop. Using exactly the same argument from Proposition 4.2, we can display the depletion loop in the following matrix form

$$\begin{array}{cccccccccc}
 0 \cdots 0 & i_1 & 0 \cdots 0 & i_2 & 0 \cdots 0 & i_3 & 0 \cdots 0 & \cdots & i_{N_1} & 0 \cdots 0 \\
 0 \cdots 0 & j_1 & 0 \cdots 0 & i_1 & 0 \cdots 0 & i_2 & 0 \cdots 0 & \cdots & i_{N_1-1} & 0 \cdots 0 \\
 0 \cdots 0 & i_{N_1} & 0 \cdots 0 & j_1 & 0 \cdots 0 & i_1 & 0 \cdots 0 & \cdots & i_{N_1-2} & 0 \cdots 0 \\
 \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
 0 \cdots 0 & i_3 & 0 \cdots 0 & i_4 & 0 \cdots 0 & i_5 & 0 \cdots 0 & \cdots & i_1 & 0 \cdots 0 \\
 0 \cdots 0 & i_2 & 0 \cdots 0 & i_3 & 0 \cdots 0 & i_4 & 0 \cdots 0 & \cdots & j_1 & 0 \cdots 0
 \end{array}$$

where $(j_1 + \sum_{l=1}^{N_1} i_l) = \alpha_1$.

Clearly, the cycle lengths of a low priority queue, are in cyclic order, $(\alpha_1 - j_1)$, $(\alpha_1 - i_{N_1})$, $(\alpha_1 - i_{N_1-1})$, $(\alpha_1 - i_{N_1-2})$, \dots , $(\alpha_1 - i_1)$. Since all low priority queues receive zero services in a depletion loop, the cycle lengths must be greater than or equal to all the elements in $\{\alpha_2, \alpha_3, \dots, \alpha_M\}$. In particular, we must have $(\alpha_1 - j_1) \geq \alpha_2$ and $(\alpha_1 - i_l) \geq \alpha_2$ for $1 \leq l \leq N_1$. Summing the $(N_1 + 1)$ inequalities, we have

$$(N_1 + 1)\alpha_1 - \left(j_1 + \sum_{l=1}^{N_1} i_l\right) \geq (N_1 + 1)\alpha_2 \quad (4.7)$$

Since $\left(j_1 + \sum_{l=1}^{N_1} i_l\right) = \alpha_1$, we have

$$(N_1 + 1)(\alpha_1 - \alpha_2) \geq \alpha_1 \quad (4.8)$$

Conversely, if inequality (4.8) is satisfied, then it is a simple matter to show that it is possible to choose $(N_1 + 1)$ non-negative integers, all no more than $(\alpha_1 - \alpha_2)$, while summing to α_1 . We give one possible choice of these $(N_1 + 1)$ numbers. Let L be the largest integer such that $\alpha_1 > L(\alpha_1 - \alpha_2)$. Clearly, $L < (N_1 + 1)$ by inequality (4.8). We let the first L of the $(N_1 + 1)$ integers be $(\alpha_1 - \alpha_2)$, the $(L + 1)$ st integer be $\left(\alpha_1 - L(\alpha_1 - \alpha_2)\right)$ and the rest be zero.

From the above set of $(N_1 + 1)$ integers, select any subset of N_1 integers. Consider the state with all low priority queues receive zero services and the N_1 1-priority queues receive services given by the elements of the chosen subset. Straightforward calculations of subsequent states show that the system is in depletion mode. Thus, the converse of the proposition is established.

□

We can rewrite (4.8) as $N_1\alpha_1 \geq (N_1 + 1)\alpha_2$. Thus, we can view the inequality either as a lower bound on α_1 or as an upper bound on α_2 . Proposition 4.3 shows that by choosing a sufficiently small value for α_2 , there always exist depletion loops. Alternatively, we can

choose a large enough value for α_2 such that the 1-priority queues do not monopolize the network. This agrees with our intuition that the parameters TRT 's are limiting factors for queue services.

Proposition 4.3 guarantees only existence of depletion loops rather than ruling out the possibility of non-depletion loops. Again by intuition, we expect that by reducing α_2 gradually, we would always come to a point where there are only depletion loops. That threshold, if exists, must clearly be less than or equal to $(N_1\alpha_1)/(N_1 + 1)$. Surprisingly, that threshold is exactly $(N_1\alpha_1)/(N_1 + 1)$. The proof requires a detailed study of the surplus and deficit of successive extended states.

4.4 Surplus and Deficit.

From now on, we shall assume without loss of generality that $p_1 = 1$, that is, the first queue has priority 1. In the light of the result from last sub-section, we can now understand the peculiar definitions of surplus and deficit given in sub-section 3.4. Recall from the proof of Proposition 4.3 that if the system is in an extended state where all 1-priority queues receive services no more than $(\alpha_1 - \alpha_2)$ and all low priority queues receive zero services, then the system will be operating in depletion mode. Equivalently, we can assert that if the system is in an extended state with zero deficit, then the system will be in depletion mode. The deficit of an extended state is a measure of “how close” the extended state is to an extended state in a depletion loop. Whether the system will operate in depletion mode depends solely on how the deficit of the extended state changes as the system evolves.

Closely related to the deficit is the surplus. We shall show in this sub-section that the difference between surplus and deficit of an extended state is constant and that both of them decrease as the system evolves.

Proposition 4.4 The difference between surplus $S_T^{(k)}$ and deficit $D_T^{(k)}$ of the k th extended state is given by

$$(D_T^{(k)} - S_T^{(k)}) = \alpha_1 - (N_1 + 1)(\alpha_1 - \alpha_2) \quad (4.9)$$

Proof

The result follows directly from the definitions of surplus and deficit. Consider the extended state $(\vec{T}^{(k)}, T_1^{(k+1)})$ where $k > 1$. Let us look at the contribution of a single component of the extended state to the quantity $(D_T^{(k)} - S_T^{(k)})$. If queue l has priority 1, then the contribution of $T_l^{(k)}$ will be $(T_l^{(k)} - (\alpha_1 - \alpha_2))$ independent of whether $T_l^{(k)}$ has a surplus or deficit. Similarly, the contribution from $T_1^{(k+1)}$ is $(T_1^{(k+1)} - (\alpha_1 - \alpha_2))$. If queue l has a lower priority, then the contribution of $T_l^{(k)}$ will just be $T_l^{(k)}$. Therefore, by summing the contributions from all components, we have

$$D_T^{(k)} - S_T^{(k)} = (T_1^{(k+1)} + \sum_{l=1}^N T_l^{(k)}) - (N_1 + 1)(\alpha_1 - \alpha_2) \quad (4.10)$$

Replacing the sum of components of an extended state by α_1 gives (4.9).

□

Proposition 4.4 shows that the difference between surplus and deficit of an extended state depends only on the system parameters N_1 , α_1 , α_2 and is thus a constant. As the system evolves, the surplus and deficit vary such that their difference is constant. If α_1 is less than $(N_1 + 1)(\alpha_1 - \alpha_2)$, then surplus exceeds deficit and vice versa. Notice that Proposition 4.4 is consistent with Proposition 4.3. From Proposition 4.4, we see that $D_T^{(k)} > 0$ if $\alpha_1 > (N_1 + 1)(\alpha_1 - \alpha_2)$; this implies that there can be no depletion loops which is the same conclusion drawn from Proposition 4.3.

The next result shows that surplus and deficit in the same pre-group “cancel” each other, leading to the monotonicity property of surplus and deficit of an extended state.

Proposition 4.5 Let $S_{pr,i}^{(k)}$ and $D_{pr,i}^{(k)}$ be the surplus and deficit of the i th pre-group in the k th extended state, $S_{po,i}^{(k)}$ and $D_{po,i}^{(k)}$ be the surplus and deficit of the i th post-group in the k th extended state, respectively. Then we have

- (i) if $S_{pr,i}^{(k)} \geq D_{pr,i}^{(k)}$, then $S_{po,i}^{(k+1)} = S_{pr,i}^{(k)} - D_{pr,i}^{(k)}$ and $D_{po,i}^{(k+1)} = 0$,
- (ii) if $D_{pr,i}^{(k)} \geq S_{pr,i}^{(k)}$, then $D_{po,i}^{(k+1)} = D_{pr,i}^{(k)} - S_{pr,i}^{(k)}$ and $S_{po,i}^{(k+1)} = 0$.

Proof

Consider the pre-groups in the k th extended state $(\vec{T}^{(k)}, T_1^{(k+1)})$ and the post-groups in the $(k+1)$ st extended state $(\vec{T}^{(k+1)}, T_1^{(k+2)})$ where $k > 1$. Clearly, the proposition is true for $i = 0$ since both the 0th pre-group in the k th extended state and the 0th post-group in the $(k+1)$ st extended state are $(T_1^{(k+1)})$. In this case, $S_{pr,0}^{(k)} = S_{po,0}^{(k+1)}$ and $D_{pr,0}^{(k)} = D_{po,0}^{(k+1)}$. We must also have either $S_{pr,0}^{(k)} = 0$ or $D_{pr,0}^{(k)} = 0$ since there is only one component in the 0th pre-group. It is now a simple matter to verify both (i) and (ii) of the proposition are satisfied for this special case.

To prove the proposition for other pre-groups, we consider, without loss of generality, the first pre-group $(T_1^{(k)}, T_2^{(k)}, \dots, T_{p_2-1}^{(k)})$ in the k th extended state and the first post-group $(T_2^{(k+1)}, T_3^{(k+1)}, \dots, T_{p_2}^{(k+1)})$ in the $(k+1)$ st extended state. Since queues 1 and p_2 are 1-priority queues, we have from Proposition 4.1 that

$$\sum_{l=1}^N T_l^{(k)} + T_1^{(k+1)} = \alpha_1 = \sum_{l=p_2}^N T_l^{(k)} + \sum_{l=1}^{p_2} T_l^{(k+1)} \quad (4.11)$$

Subtracting $(\sum_{l=p_2}^N T_l^{(k)}) + T_1^{(k+1)} + (\alpha_1 - \alpha_2)$ from both sides of the equation, we have

$$T_1^{(k)} - (\alpha_1 - \alpha_2) + \sum_{l=2}^{p_2-1} T_l^{(k)} = \sum_{l=2}^{p_2-1} T_l^{(k+1)} + T_{p_2}^{(k+1)} - (\alpha_1 - \alpha_2) \quad (4.12)$$

The left hand side of (4.12) is equal to $(D_{pr,1}^{(k)} - S_{pr,1}^{(k)})$ and the right hand side is equal to $(D_{po,1}^{(k+1)} - S_{po,1}^{(k+1)})$ following the same argument in Proposition 4.4. Therefore,

$$D_{pr,1}^{(k)} - S_{pr,1}^{(k)} = D_{po,1}^{(k+1)} - S_{po,1}^{(k+1)} \quad (4.13)$$

It is clear that the above result holds true for $p_2 = 2$ as well if we regard summation as zero when the lower summation index exceeds the upper index. In this trivial case, we have $T_2^{(k+1)} = T_1^{(k)}$ which implies either

$$S_{pr,1}^{(k)} = S_{po,1}^{(k+1)} \geq D_{pr,1}^{(k)} = D_{po,1}^{(k+1)} = 0$$

or

$$D_{pr,1}^{(k)} = D_{po,1}^{(k+1)} \geq S_{pr,1}^{(k)} = S_{po,1}^{(k+1)} = 0$$

In both cases, the proposition can be verified easily.

We now prove the proposition for the non-trivial case where $p_2 > 2$. Consider the case where $p_2 > 2$ and $S_{pr,1}^{(k)} \geq D_{pr,1}^{(k)}$. Then we have

$$D_{pr,1}^{(k)} - S_{pr,1}^{(k)} = \sum_{l=1}^{p_2-1} T_l^{(k)} - (\alpha_1 - \alpha_2) \leq 0 \quad (4.14)$$

If there exists $l \in \{2, 3, \dots, p_2 - 1\}$ such that $T_l^{(k+1)} > 0$, let n be the minimum of these integers. Then

$$T_n^{(k+1)} = TRT_n - C_n^{(k)} = TRT_n - \left(\alpha_1 - \sum_{l=1}^{n-1} T_l^{(k)} \right) \quad (4.15)$$

Using (4.14), we have

$$T_n^{(k+1)} \leq TRT_n - \alpha_2 \leq 0 \quad (4.16)$$

which contradicts $T_n^{(k+1)} > 0$. Therefore, $T_l^{(k+1)} = 0$ for $2 \leq l \leq (p_2 - 1)$. Thus, $D_{po,1}^{(k+1)}$ is equal to the deficit of $T_{p_2}^{(k+1)}$. We also know that $S_{po,1}^{(k+1)}$ is equal to the surplus of $T_{p_2}^{(k+1)}$ since queue p_2 is the only 1-priority queue associated with the first post-group. Moreover, (4.13) and (4.14) show that $S_{po,1}^{(k+1)} \geq D_{po,1}^{(k+1)}$. Therefore, $D_{po,1}^{(k+1)}$ must be zero and so $S_{po,1}^{(k+1)} = S_{pr,1}^{(k)} - D_{pr,1}^{(k)}$ from (4.13). Hence, part (i) of the proposition is established.

Consider the case where $S_{pr,1}^{(k)} \leq D_{pr,1}^{(k)}$. If $S_{po,1}^{(k+1)} > 0$, then $T_{p_2}^{(k+1)} < (\alpha_1 - \alpha_2)$. Also from (4.13), we must have $D_{po,1}^{(k+1)} > 0$. This means that there exists $l \in \{2, 3, \dots, p_2 - 1\}$ such that $T_l^{(k+1)} > 0$; let n be the maximum of these integers. Then

$$T_{p_2}^{(k+1)} = \alpha_1 - C_{p_2}^{(k)} = \alpha_1 - (TRT_n - \sum_{l=n}^{p_2-1} T_l^{(k)}) \geq \alpha_1 - TRT_n \geq (\alpha_1 - \alpha_2) \quad (4.17)$$

which contradicts $T_{p_2}^{(k+1)} < (\alpha_1 - \alpha_2)$. Therefore, $S_{po,1}^{(k+1)} = 0$. Using (4.13) will establish part (ii) of the proposition. □

The implications of Proposition 4.5 are important. The proposition implies that $D_{pr,i}^{(k)} \geq D_{po,i}^{(k+1)}$ and $S_{pr,i}^{(k)} \geq S_{po,i}^{(k+1)}$ for $0 \leq i \leq N_1$. Therefore,

$$D_T^{(k)} = \sum_{i=0}^{N_1} D_{pr,i}^{(k)} \geq \sum_{i=0}^{N_1} D_{po,i}^{(k+1)} = D_T^{(k+1)} \quad (4.18)$$

and

$$S_T^{(k)} = \sum_{i=0}^{N_1} S_{pr,i}^{(k)} \geq \sum_{i=0}^{N_1} S_{po,i}^{(k+1)} = S_T^{(k+1)} \quad (4.19)$$

We can further conclude from Proposition 4.5 that if there is a pre-group in the k th extended state with non-zero surplus and non-zero deficit, then

$$D_T^{(k)} > D_T^{(k+1)} \quad (4.20)$$

and

$$S_T^{(k)} > S_T^{(k+1)} \quad (4.21)$$

This monotonicity property is the key to characterize the loops of a general system.

If $D_T^{(k)} > 0$ and $S_T^{(k)} > 0$ but no pre-group has both non-zero surplus and non-zero deficit, then $D_T^{(k+1)} = D_T^{(k)}$ and $S_T^{(k+1)} = S_T^{(k)}$. In this case, we cannot guarantee strict decrement of deficit and surplus. The next result deals with this case.

Proposition 4.6 Let $V_s^{(k)}$ be the set of indices of all pre-groups in the k th extended state with non-zero surplus and $V_d^{(k)}$ be the set of indices of all pre-groups in the k th extended state with non-zero deficit. Define the directional distance between two subsets V, U of $\{0, 1, \dots, N_1\}$ by

$$d(V, U) = \min_{r \in V, s \in U} ((r - s) \bmod (N_1 + 1))$$

If $d(V_d^{(k)}, V_s^{(k)}) > 0$, then there exists a positive integer m such that $d(V_d^{(k)}, V_s^{(k)}) > d(V_d^{(k+m)}, V_s^{(k+m)})$.

Proof

Let us use modulo $(N_1 + 1)$ arithmetic for indices of pre-groups and post-groups throughout this proof. We first notice that the directional distance function is not symmetric, that is, $d(U, V) \neq d(V, U)$. However, $d(V, U) \geq 0$ and $d(V, U) = 0$ if and only if the sets V and U have at least one common element. If $d(V_d^{(k)}, V_s^{(k)}) > 0$, then no pre-groups in the k th extended state has both non-zero surplus and non-zero deficit. Let $i \in V_s^{(k)}$ and $j \in V_d^{(k)}$ such that $(j - i) = d(V_d^{(k)}, V_s^{(k)}) > 0$.

Since $S_{pr,i}^{(k)} > 0$ and $D_{pr,i}^{(k)} = 0$, then from Proposition 4.5, we have $S_{po,i}^{(k+1)} = S_{pr,i}^{(k)}$. In other words, the surplus of $T_{p_i}^{(k)}$ is equal to the surplus of $T_{p(i+1)}^{(k+1)}$. Thus, $(i + 1) \in V_s^{(k+1)}$.

Since $D_{pr,j}^{(k)} > 0$ and $S_{pr,j}^{(k)} = 0$, then we have $D_{po,j}^{(k+1)} = D_{pr,j}^{(k)}$. In general, the deficit of the j th pre-group will be distributed among the components of the j th post-group after one state transition. Thus, either $j \in V_d^{(k+1)}$ or $(j + 1) \in V_d^{(k+1)}$ or both. Furthermore, if $j \notin V_d^{(k+1)}$, then the group leader of the $(j + 1)$ st pre-group in the $(k + 1)$ st extended state must have a non-zero deficit.

We now show that if there is a 2-priority queue associated with the j th pre-group and that the group leader of this pre-group has a non-zero deficit, then $j \in V_d^{(k+1)}$. Let n be the smallest index of the 2-priority queues associated with the j th pre-group. If

$j \notin V_d^{(k+1)}$, then we have

$$C_n^{(k)} = \alpha_1 - \sum_{l=p_j}^{n-1} T_l^{(k)} \leq \alpha_1 - T_{p_j}^{(k)}$$

However, $(\alpha_1 - T_{p_j}^{(k)}) < \alpha_2$ because the group leader is assumed to have a non-zero deficit. Thus, $C_n^{(k)} < \alpha_2$ and so $T_n^{(k+1)} > 0$, contradicting the hypothesis that $j \notin V_d^{(k+1)}$. Therefore, we must have $j \in V_d^{(k+1)}$.

With the above results, we can proceed to prove the proposition by contradiction. First of all, it is clear that $d(V_d^{(k)}, V_s^{(k)})$ cannot be larger than $d(V_d^{(k+1)}, V_s^{(k+1)})$ since $(i+1) \in V_s^{(k+1)}$ and either j or $(j+1)$ belong to $V_d^{(k+1)}$. In other words, the directional distance, if positive, cannot increase. Let us assume $d(V_d^{(k)}, V_s^{(k)}) = d(V_d^{(k+m)}, V_s^{(k+m)})$ for all positive m , then we must have $j \notin V_d^{(k+1)}$ and so the group leader of the $(j+1)$ st pre-group in the $(k+1)$ st extended state must have a non-zero deficit. After another state transition, we must now have $(j+1) \notin V_d^{(k+2)}$. This means that there is no 2-priority queue associated with the $(j+1)$ st pre-group. Continuing this argument would show that there is no 2-priority queue in the system which is a contradiction. Thus, there exists a positive integer m such that $d(V_d^{(k)}, V_s^{(k)}) > d(V_d^{(k+m)}, V_s^{(k+m)})$. In other words, the directional distance, if positive, must strictly decrease after a finite number of state transitions.

□

Combining Propositions 4.5 and 4.6, we see that surplus and deficit of an extended state, if both positive, must strictly decrease as the system evolves. We state this as a theorem.

Theorem 4.7 If $D_T^{(k)} > 0$ and $S_T^{(k)} > 0$, then there exists a positive integer m such that $D_T^{(k)} > D_T^{(k+m)}$ and $S_T^{(k)} > S_T^{(k+m)}$.

Proof

If $d(V_d^{(k)}, V_s^{(k)}) = 0$, then there exist i such that $S_{pr,i}^{(k)} > 0$ and $D_{pr,i}^{(k)} > 0$. Then by Proposition 4.5, we will have $D_T^{(k)} > D_T^{(k+1)}$ and $S_T^{(k)} > S_T^{(k+1)}$.

If $d(V_d^{(k)}, V_s^{(k)}) > 0$, then the directional distance must strictly decrease after a finite number of state transitions. This means the directional distance must eventually be zero. From above, we must have strict decrement in both surplus and deficit one state transition after the directional distance has become zero.

□

4.5 Characterization of Depletion and Non-depletion Systems.

The discussion on surplus and deficit of an extended state provides the basis for characterization of recurrent sets of a general asymmetric system. A corollary of Theorem (4.7) is that all loops are either of the depletion type or all are of the non-depletion type. This is because if surplus exceeds deficit, the deficit will eventually become zero and the system will operate in depletion mode. Otherwise if deficit exceeds surplus, the deficit will never reach zero and hence the system will be in non-depletion mode.

Corollary 4.8 The system will operate in depletion mode if and only if $(N_1 + 1)(\alpha_1 - \alpha_2) \geq \alpha_1$.

Proof

If $(N_1 + 1)(\alpha_1 - \alpha_2) \geq \alpha_1$, then from Proposition 4.4 the surplus is no less than the deficit of the same extended state. From Theorem 4.7, the deficit of an extended state will eventually become zero. Thus, the system will operate in depletion mode.

If $(N_1 + 1)(\alpha_1 - \alpha_2) < \alpha_1$, then from Proposition 4.3 the system will not operate in depletion mode.

□

The above corollary is an obvious consequence of Theorem 4.7 and other previous results. It turns out that there is another non-obvious corollary that allows us to construct the non-depletion loops of a system from the depletion loops of another system with different parameters.

Corollary 4.9 Define a decreasing sequence of numbers in terms of α_k and N_k by

$$f_1 = \alpha_1 \tag{4.22a}$$

$$f_{k+1} = f_k - (\alpha_k - \alpha_{k+1}) \left(1 + \sum_{l=1}^k N_l\right) \tag{4.22b}$$

for $k = 1, \dots, M - 1$.

Let f_n be the last non-negative number in this sequence. Then queues with priority lower than n will not receive any service. Other queues will receive an average service per cycle of

$$\bar{S}_k = (\alpha_k - \alpha_n) + \frac{f_n}{1 + \sum_{l=1}^n N_l} \tag{4.23}$$

for $k = 1, 2, \dots, n$.

The average cycle length is given by

$$\bar{C} = \alpha_k - \bar{S}_k = \sum_{l=1}^n N_l \alpha_l / \left(1 + \sum_{l=1}^n N_l\right) \tag{4.24}$$

for $k = 1, 2, \dots, n$.

Proof

Consider a non-depletion system. From Theorem 4.7, we deduce that surplus of an extended state must be zero within a loop. If surplus is zero, then all 1-priority queues will receive services no less than $(\alpha_1 - \alpha_2)$. In this case, we can let $U_i^{(k)} = T_i^{(k)} - (\alpha_1 - \alpha_2)$

if queue i has priority 1 and $U_i^{(k)} = T_i^{(k)}$ if queue i has lower priority. We can then rewrite the Markov relations (3.6) and (3.7) as

$$C_i^{(k)} - N_1(\alpha_1 - \alpha_2) = \sum_{j=i}^N U_j^{(k)} + \sum_{j=1}^{i-1} U_j^{(k+1)} \quad (4.25)$$

and for 1-priority queues,

$$U_i^{(k+1)} = (\alpha_1 - (N_1 + 1)(\alpha_1 - \alpha_2)) - (C_i^{(k)} - N_1(\alpha_1 - \alpha_2)) \quad (4.26a)$$

and for low priority queues,

$$U_i^{(k+1)} = \max \left((TRT_i - N_1(\alpha_1 - \alpha_2)) - (C_i^{(k)} - N_1(\alpha_1 - \alpha_2)), \quad 0 \right) \quad (4.26b)$$

The above equations show that the vector $\vec{U}^{(k)} = (U_1^{(k)}, U_2^{(k)}, \dots, U_N^{(k)})$ corresponds to the system with α_1 modified as $(\alpha_1 - (N_1 + 1)(\alpha_1 - \alpha_2))$ and for $2 \leq k \leq M$, α_k is modified as $(\alpha_k - N_1(\alpha_1 - \alpha_2))$. We notice that

$$\alpha_1 - (N_1 + 1)(\alpha_1 - \alpha_2) = \alpha_2 - N_1(\alpha_1 - \alpha_2) = f_2 \quad (4.27)$$

Thus, we have transformed the system to a new equivalent system consisting of $(N_1 + N_2)$ 1-priority queues with TRT equal to f_2 and N_k $(k - 1)$ -priority queues with TRT equal to $f_2 - (\alpha_2 - \alpha_k)$ for $3 \leq k \leq M$.

The new system with modified parameters is equivalent to the original system in the sense that we can construct the loops of the original system directly from the loops of the new system by adding $(\alpha_1 - \alpha_2)$ to the services of the original 1-priority queues. If the equivalent system has non-depletion loops, we can use the same argument to further transform the equivalent system to another one with $(N_1 + N_2 + N_3)$ 1-priority queues and largest TRT parameter equal to f_3 . We can repeat this argument $(M - 1)$ times or until

the transformed maximum TRT is negative. At that point, we will have a symmetric or depletion system.

After we have transformed the system to either a symmetric or depletion system, we can use the same steps discussed in sub-section 4.2 to compute the average service per cycle of each queue with non-zero throughput as $f_n/(1+\sum_{l=1}^n N_l)$. Recall that at each stage of the transformation, we have subtracted from the services of 1-priority queues the difference between the largest and the second largest TRT 's. In the first stage, this difference is $(\alpha_1 - \alpha_2)$. In the second stage, the difference is $((\alpha_2 - N_1(\alpha_1 - \alpha_2)) - (\alpha_3 - N_1(\alpha_1 - \alpha_2)))$ which is $(\alpha_2 - \alpha_3)$. Continuing, we see that at the final stage of the transformation, we have subtracted $(\alpha_k - \alpha_n)$ from the services of the original k -priority queues where $1 \leq k \leq n$. Thus, the k -priority queues in the original system receive an average service per cycle of

$$\bar{S}_k = (\alpha_k - \alpha_n) + \frac{f_n}{1 + \sum_{l=1}^n N_l}$$

for $k = 1, 2, \dots, n$.

The average cycle time of queue i with non-zero throughput is the difference between the TRT and average service per cycle of that queue. It can be shown easily by induction that

$$f_k = \alpha_k - \sum_{l=1}^k N_l(\alpha_l - \alpha_k) \quad (4.28)$$

for $1 \leq k \leq M$. Using (4.23) and (4.28), we have

$$\bar{C} = \alpha_k - \bar{S}_k = \sum_{l=1}^n N_l \alpha_l / \left(1 + \sum_{l=1}^n N_l\right)$$

□

The above corollary is of major significance because it allows us to calculate the bandwidth allocated to each queue from the system parameters N_k and α_k . The bandwidth

allocation is shown to be linearly increasing in the TRT assigned. Two queues with the same TRT are given the same bandwidth and a queue is allocated more bandwidth than a queue with a smaller TRT . Thus, we have justified the assignment of priority based on the TRT parameters.

Depending on the given parameters, some queues may not receive any service at all. This is in contrast to the service-limiting schemes with fixed quotas where no queues will be deprived of service. It is also interesting to note that the allocation of bandwidth is independent of the relative positions of the queues. In general, the transient state transitions, that is, how long and in what way does the system enter a loop, depend on the relative positions of the queues. Queueing behaviour of a non-heavily loaded system is also dependent on relative queue positions.

4.6 Remarks on More General Systems.

We have derived some very interesting results in previous sub-sections. However, these results apply only to heavily loaded systems with constant type I services, constant walk times and negligible overflows. We would like to comment on more general systems with these assumptions relaxed.

First of all, we want to point out that all the results we have derived apply to continuous time systems as well. We have relied on the assumption that the system is time-slotted in only one occasion, namely, the monotonicity of deficit and surplus implies one of them converging to zero. This is used in the proof of Corollaries (4.8) and (4.9). It can be shown that even in a continuous time system, the deficit or surplus of an extended state will become zero after a finite number of state transitions. The proof is based on the observation that the number of pre-groups with positive surplus will strictly decrease if both surplus or deficit of an extended state remain positive. Since surplus of a pre-group,

if non-zero, is non-vanishing, we must have convergence to zero of either surplus or deficit after a finite number of state transitions.

The second issue we want to address is the assumption of constant walk times. If we assume a logical ring is maintained, the assumption of constant walk times is valid for many real systems, such as Expressnet [12]. For other systems, we may have to consider the variances of token transmission time, propagation delay and bit delay at a station, as discussed in [19]. In any event, if the variances of the walk times are small compared with their means, we can essentially regard the system as having constant walk times perturbed by “noise”. Clearly, the system characteristics depend very much on the magnitude of the perturbation. For extremely small perturbation, we expect that Corollary 4.9 still applies. As the perturbation increases, we would expect that some low priority queues, originally depleted of service, may have non-zero throughputs in a probabilistic sense by virtual of the fluctuations in the walk times. The detailed analysis can be carried out using equations (3.1) and (3.2b), treating the walk times as random variables.

In many ways, the effects arise from non-constant type I services are similar to that from non-constant walk times. However, there is one major difference. Non-constant walk times can usually be treated as independent random variables, among different queues and different cycles. Depending on the nature of the application, services of a given type I queue may be essentially uncorrelated (*e.g.* interrupts to a peripheral device from a processor) or highly correlated (*e.g.* voice sources). For the case of uncorrelated services, we can proceed along the same avenue for non-constant walk times. For voice applications where the correlation can be modelled as a two-state Markov chain with a small probability of jumping from one state to the other, we can treat the system as being “quasi-static”. Thus, we assume the system is subject to infrequent perturbations that create surplus and deficit in an extended state; after such perturbations, the system is allowed to attain the steady state before the next perturbation. Hence, it is clear that the analysis of transient

state transitions, which we did not investigate fully in this report, is also very important.

The overflow effect, although similar to that arisen from non-constant walk times and non-constant type I services, has a distinct flavor. It is dependent on the statistics of type II packet lengths. The rigorous approach to study this effect is to modify equation (3.2b) to accommodate overflow. Unfortunately, the rigorous approach leads to rather intractable non-linear system equations.

Finally, we would like to consider the removal of the heavy load assumption. If we remove the heavy load assumption, the vector of service times in a cycle is no longer a Markov chain. Instead, we have to consider the imbedded population process. Again, the rigorous approach leads to completely intractable mathematics. There have been many attempts, with different degrees of success, to derive bounds and approximations for token-passing systems with limited service disciplines but it is beyond the scope of this report to survey all such work. Heuristically, the heavy load situation is a "worst case" scenerio where the competitions among different queues are most prominent, for example, we expect that a queue will always be stable if its offered load is less than the throughput calculated from the heavy load analysis. Thus, we can obtain good bench marks for selecting system parameters based on the results derived under the heavy load assumption.

4.7 Numerical Examples.

We shall work through two numerical examples in this sub-section to demonstrate how we can apply the analytic results derived in this section.

Example 4.1 Let there be three asynchronous classes of traffic with $N_1 = 4$, $N_2 = 6$, $N_3 = 9$ and $\alpha_1^{(org)} = 105$, $\alpha_2^{(org)} = 100$, $\alpha_3^{(org)} = 95$ (slots). Let the total walk time $W = 5$ (slots). Under the heavy load assumption,

- find the average cycle length, the average service time per cycle and throughput of a queue in each asynchronous class if there are no synchronous queues,
- repeat a) if there are 4 identical synchronous queues each using 5 slots in each cycle,
- find the throughput of a queue in each asynchronous class for $N_2 > 6$ assuming synchronous traffic given in b),
- find the throughput of a queue in each asynchronous class for $N_1 > 4$ assuming synchronous traffic given in b).

Solution:

$$\begin{aligned} \text{a) } \alpha_1^{(eq)} &= \alpha_1^{(org)} - W = 105 - 5 = 100, \\ \alpha_2^{(eq)} &= \alpha_2^{(org)} - W = 100 - 5 = 95, \\ \alpha_3^{(eq)} &= \alpha_3^{(org)} - W = 95 - 5 = 90. \end{aligned}$$

Using (4.22a) and (4.22b), we have

$$\begin{aligned} f_1 &= \alpha_1^{(eq)} = 100, \\ f_2 &= f_1 - (\alpha_1^{(eq)} - \alpha_2^{(eq)})(1 + N_1) = 100 - (100 - 95)(1 + 4) = 75, \\ f_3 &= f_2 - (\alpha_2^{(eq)} - \alpha_3^{(eq)})(1 + N_1 + N_2) = 75 - (95 - 90)(1 + 4 + 6) = 20. \end{aligned}$$

All queues have non-zero throughputs.

Using (4.23), we have

$$\bar{S}_1 = (\alpha_1^{(eq)} - \alpha_3^{(eq)}) + f_3 / (1 + N_1 + N_2 + N_3) = (100 - 90) + 20 / (1 + 4 + 6 + 9) = 11,$$

$$\bar{S}_2 = (\alpha_2^{(eq)} - \alpha_3^{(eq)}) + f_3 / (1 + N_1 + N_2 + N_3) = (95 - 90) + 20 / (1 + 4 + 6 + 9) = 6,$$

$$\bar{S}_3 = (\alpha_3^{(eq)} - \alpha_3^{(eq)}) + f_3 / (1 + N_1 + N_2 + N_3) = (90 - 90) + 20 / (1 + 4 + 6 + 9) = 1.$$

Using (4.24), we have $\bar{C}^{(eq)} = \alpha_1^{(eq)} - \bar{S}_1 = 100 - 11 = 89.$

Thus, $\bar{C}^{(org)} = \bar{C}^{(eq)} + W = 89 + 5 = 94.$

Let ρ_k be the throughput of a queue in the k th asynchronous class. Then

$$\rho_1 = \bar{S}_1 / \bar{C}^{(org)} = 11 / 94,$$

$$\rho_2 = \bar{S}_2 / \bar{C}^{(org)} = 6 / 94,$$

$$\rho_3 = \bar{S}_3 / \bar{C}^{(org)} = 1 / 94.$$

b) $\alpha_1^{(eq)} = \alpha_1^{(org)} - (W + H) = 105 - (5 + 20) = 80,$

$$\alpha_2^{(eq)} = \alpha_2^{(org)} - (W + H) = 100 - (5 + 20) = 75,$$

$$\alpha_3^{(eq)} = \alpha_3^{(org)} - (W + H) = 95 - (5 + 20) = 70.$$

$$f_1 = 80,$$

$$f_2 = 80 - (80 - 75)(1 + 4) = 55,$$

$$f_3 = 55 - (75 - 70)(1 + 4 + 6) = 0.$$

Class 3 queues have zero throughputs.

$$\bar{S}_1 = (80 - 75) + 55 / (1 + 4 + 6) = 10,$$

$$\bar{S}_2 = (75 - 75) + 55 / (1 + 4 + 6) = 5,$$

$$\bar{S}_3 = 0.$$

$$\bar{C}^{(eq)} = (80 - 10) = 70$$

$$\bar{C}^{(org)} = 70 + (5 + 20) = 95.$$

$$\rho_1 = 10 / 95,$$

$$\rho_2 = 5 / 95,$$

$$\rho_3 = 0.$$

c) $\alpha_1^{(eq)} = 80,$

$$\alpha_2^{(eq)} = 75,$$

$$\alpha_3^{(eq)} = 70.$$

$$f_1 = 80,$$

$$f_2 = 55,$$

$$f_3 < 0.$$

Class 3 queues have zero throughputs.

$$\bar{S}_1 = 5 + 55/(5 + N_2),$$

$$\bar{S}_2 = 55/(5 + N_2),$$

$$\bar{S}_3 = 0.$$

$$\bar{C}^{(eq)} = 75 - 55/(5 + N_2),$$

$$\bar{C}^{(org)} = 100 - 55/(5 + N_2).$$

$$\rho_1 = (16 + N_2)/(89 + 20N_2),$$

$$\rho_2 = 11/(89 + 20N_2),$$

$$\rho_3 = 0.$$

We see that as $N_2 \rightarrow \infty$, $\rho_1 \rightarrow 0.05$ and $\rho_2 \rightarrow 0$.

d) $\alpha_1^{(eq)} = 80,$

$$\alpha_2^{(eq)} = 75,$$

$$\alpha_3^{(eq)} = 70.$$

$$f_1 = 80,$$

$$f_2 = 75 - 5N_1,$$

$$f_3 < 0.$$

Class 2 queues have zero throughputs if $N_1 \geq 15$.

Class 3 queues have zero throughputs.

$$\bar{S}_1 = 5 + (75 - 5N_1)/(7 + N_1),$$

$$\bar{S}_2 = (75 - 5N_1)/(7 + N_1) \text{ for } 4 < N_1 \leq 15,$$

$$\bar{S}_3 = 0.$$

$$\bar{C}^{(eq)} = 75 - (75 - 5N_1)/(7 + N_1),$$

$$\bar{C}^{(org)} = 100 - (75 - 5N_1)/(7 + N_1).$$

$$\rho_1 = 22/(125 + 21N_1),$$

$$\rho_2 = (15 - N_1)/(125 + 21N_1),$$

$$\rho_3 = 0.$$

Example 4.2 K identical stations communicate with each other on a token-passing network that has a channel bandwidth of 10 Mbps and a round-trip propagation delay of $20\mu s$. Each station has one synchronous queue supporting a 64 kbps voice channel and three asynchronous queues requiring minimum bandwidths of 50, 40 and 10 kbps. Voice packets are 1280 bits long. Design an integrated services network supporting a maximum number of stations based on the THT/TRT mechanism.

Solution:

We are required to select the design parameters THT , $\alpha_1^{(org)}$, $\alpha_2^{(org)}$ and $\alpha_3^{(org)}$ to maximize K while satisfying the bandwidth requirements and maintaining voice quality. Let us formulate the optimization problem.

First of all, we shall assume all walk times are constant and the total walk time is given by the round-trip propagation delay, that is, $W = 20$. All time units will be in μs .

We now look at the requirements of a voice channel. The time to transmit a voice packet is the size of a voice packet divided by the channel bandwidth which is 128. Thus, we let $THT = 128$ for all station. We shall assume that at steady state, all stations transmit a voice packet in each cycle. Hence, $H_i = 12.8$ and $H = KH_i = 128K$. For voice application, overdue packets that are not delivered before the next packet arrival will be discarded. To maintain voice quality, we impose a maximum access delay requirement that is equal to the inter-arrival time of voice packets. The inter-arrival time can be obtained by dividing the voice packet size by the voice bandwidth. Thus, we have $R_{max} = 20,000 \geq \alpha_1^{(org)}$.

The bandwidth allocated to a queue in each asynchronous class is given by the throughput ρ_i times the channel bandwidth B . If B_i is the minimum bandwidth requirement of class i asynchronous queues, we must have $\rho_i B \geq B_i$ for $1 \leq i \leq 3$. The throughput is given by

$$\rho_i = \frac{\bar{S}_i}{\bar{C}^{(org)}} = \frac{\alpha_i^{(eq)} - \bar{C}^{(eq)}}{\bar{C}^{(org)}} = \frac{\alpha_i^{(org)} - \bar{C}^{(org)}}{\bar{C}^{(org)}} = \frac{\alpha_i^{(org)}}{\bar{C}^{(org)}} - 1$$

Thus, we must have

$$\frac{\alpha_i^{(org)}}{(1 + B_i/B)} \geq \bar{C}^{(org)}$$

Using (4.24), we have

$$\frac{\alpha_i^{(org)}}{(1 + B_i/B)} \geq \frac{\sum_{j=1}^3 K \alpha_j^{(eq)}}{1 + \sum_{j=1}^3 K} + (H + W) = \frac{K(THT) + W + K \sum_{j=1}^3 \alpha_j^{(org)}}{1 + 3K}$$

Rewriting the above inequality, we have

$$\frac{r_i - W}{(THT + \sum_{j=1}^3 l_j r_j) - 3r_i} \geq K$$

where $l_i \equiv 1 + B_i/B$ and $r_i \equiv \alpha_i^{(org)}/l_i$ provided $r_i > W$ and $(THT + \sum_{j=1}^3 l_j r_j) > 3r_i$ for $1 \leq i \leq 3$.

The formal optimization problem is to maximize K with respect to $\{\alpha_i^{(org)}\}$ subject to the constraints

$$\frac{r_i - W}{(THT + \sum_{j=1}^3 l_j r_j) - 3r_i} \geq K \quad (4.29)$$

$$R_{max} \geq \alpha_1^{(org)} > \alpha_2^{(org)} > \alpha_3^{(org)} \quad (4.30)$$

To solve the optimization problem, we first observe that if we keep $F \equiv \sum_{j=1}^3 l_j r_j$ fixed, then the most severe constraint given by (4.29) is the one with the smallest r_i . Thus, the

optimization problem becomes that of maximizing the minimum of r_i for a fixed $\sum_{j=1}^3 l_j r_j$. Clearly, the optimum point is at $r_1 = r_2 = r_3$, that is, $\alpha_i^{(org)}$ is proportional to $(1 + B_i/B)$. Thus, we can rewrite (4.29) as

$$\frac{(F/\sum_{j=1}^3 l_j) - W}{(THT + F) - 3F/\sum_{j=1}^3 l_j} \geq K \quad (4.31)$$

From (4.31), we see that to maximize K , we must maximize F . To maximize F , we maximize $\alpha_1^{(org)}$. Thus, the maximum value of K is obtained by letting $\alpha_1^{(org)} = R_{max}$ with $\alpha_2^{(org)}$ and $\alpha_3^{(org)}$ selected according to the proportionality criterion.

We can now substitute the numerical values. For rounded numbers, we let $R_{max} = 19,095$ instead of 20,000. Thus, we have

$$THT = 128,$$

$$\alpha_1^{(org)} = 19,095,$$

$$\alpha_2^{(org)} = 19,076,$$

$$\alpha_3^{(org)} = 19,019,$$

$$K_{max} = 59.$$

The reader can check the above figures by calculating the bandwidth allocated to a queue in each class using those figures. We caution the reader once again that the above figures are obtained under the heavy load assumption with constant type I traffic, constant walk times and negligible overflows. Practical design may require additional considerations that are beyond the scope of this report.

5. Summary.

Integrated services LAN is a relatively new area of interest. Severe delay constraints required by many real-time applications cannot be met by LAN protocols originally designed for conventional computer communication. A class of integrated access protocols are based on token-passing with control on the cycle lengths. We have examined one such protocol, which is proposed in the IEEE 802.4 token bus standard. In the IEEE 802.4 standard, timers are used to control channel access of different classes of traffic. From a practical point of view, the timing mechanism employed in the standard is both robust and easy to implement.

We used a rather general cyclic queueing model to investigate the throughput performance of different classes of traffic in the IEEE 802.4 protocol under heavy load with constant synchronous traffic, constant walk times and negligible overflows. The problem is essentially that of characterizing state transitions of a deterministic Markov chain. We have shown that access delay of a type I queue is bounded by the largest TRT parameter of the system. We have also found, using the monotonicity property of surplus and deficit, a simple characterization of the steady state service pattern in a cycle. It was found that under heavy load, the bandwidth allocated to an asynchronous queue is an increasing linear function of its TRT parameter. In this respect, a queue with larger TRT indeed have a higher priority.

The results we have obtained under the heavy load assumption provide guidelines for practical design. We have also discussed very briefly on more general systems that deviate from the basic assumptions made in this report. Finally, we worked through two numerical examples. The first example is a direct application of the results derived in this report. The second is a practical design example, aiming to illustrate the usefulness of the results presented.

References.

- [1] F. A. Tobagi, "Multiaccess Protocols and Real-time Communications in Local Area Networks," Department of Electrical Engineering, Stanford University, Stanford, California, 1986.
- [2] R. Rom and F. A. Tobagi, "Message-Based Priority Functions in Local Multiaccess Communication Systems," *Computer Networks*, vol. 5, no. 4, pp. 273-286, July 1981.
- [3] D. Towsley and G. Venkatesh, "Window Random Access Protocols for Local Computer Networks," *IEEE Transactions on Computers*, vol. C-31, no. 8, pp. 715-722, August 1982.
- [4] M. L. Molle and L. Kleinrock, "Virtual Time CSMA: Why Two Clocks are Better than One," *IEEE Transactions on Communications*, vol. COM-33, no. 9, pp 919-933, September 1985.
- [5] M. L. Molle and M.-W. Wu, "Prioritized-Virtual-Time CSMA: Head-of-the-line Priority Classes without added Overhead," *Proceedings ICC'85*.
- [6] J. F. Kurose and M. Schwartz, "A Family of Window Protocols for Time Constrained Applications in CSMA Networks," *INFOCOM'83*, San Diego, April 1983.
- [7] N. F. Maxemchuk, "A Variation of CSMA/CD That Yields Movable TDM Slots in Integrated Voice/Data Local Networks," *Bell System Technical Journal*, vol. 61, no.7, pp. 1527-1550, September 1982.
- [8] T. A. Gonsalves, "Comparative Performance of Broadcast Bus Local Area Networks with Voice and Data Traffic," Ph. D. Dissertation, Department of Electrical Engineering, Stanford University, Stanford, California, June 1986.
- [9] T. A. Gonsalves, "Packet-Voice Communication on an Ethernet Local Network: an

- Experimental Study," in *ACM Sigcomm, Symposium on Communications Architectures and Protocols*, Austin, Texas, pp. 178–185, March 1983.
- [10] F. A. Tobagi and N. Gonzalez-Cawley, "On CSMA-CD Local Networks and Voice Communication," *INFOCOM'82*, Las Vegas, Nevada, April 1982.
 - [11] M. M. Nassehi, "Channel Access Schemes and Fiber Optic Configurations for Integrated Services Local Area Network," Ph.D. Dissertation, Department of Electrical Engineering, Stanford University, Stanford, California 1986.
 - [12] F. A. Tobagi, F. Borgonovo, and L. Fratta, "EXPRESS-NET: A High-Performance Integrated-Services Local Area Network," *IEEE Journal on Selected Areas in Communication*, vol. SAC-1, no. 5, pp. 898–913, November 1983.
 - [13] M. Fine and F. A. Tobagi, "Packet Voice on a Local Area Network with Round Robin Service," *IEEE Transactions on Communications*, vol. COM-34, no. 9, pp. 906–915, September 1986.
 - [14] J. O. Limb and L. E. Flamm, "A Distributed Local Area Network Packet Protocol for Combined Voice and Data Transmission," *IEEE Journal on Selected Areas in Communications*, vol. SAC-1, no. 5, pp. 926–934, November 1983.
 - [15] J. W. Mark and J. O. Limb, "Integrated Voice/Data Services on Fasnnet," *Bell System Technical Journal*, vol. 63, no. 2, pp. 307–337, February 1984.
 - [16] J. W. Mark, J. A. Field, G. B. Agnew, and J. W. Wong, "Wlnet: A High Performance Integrated Services Local Area Network," *GLOBECOM'85*.
 - [17] A. A. Lazar, A. Patir, T. Takahashi, and M. El Zarki, "MAGNET: Columbia's Integrated Network Testbed," *IEEE Journal on Selected Areas in Communications*, vol. SAC-3, no. 6, pp. 859–871, November 1985.
 - [18] *ANSI/IEEE std 802.4-1985* "Token-pasing Bus Access Method and Physical Layer Specifications," The Institute of Electrical and Electronic Engineers, Inc., New York,

1985.

- [19] S. R. Sachs, K.-L. Kan, and J. A. Silvester, "Token-bus Protocol Performance Analysis and Comparison with other LAN Protocols," *GLOBECOM'85*.
- [20] M. J. Johnson, "Fairness of Channel Access for Non-Time-Critical Traffic Using the FDDI Token Ring Protocol," Research Institute for Advanced Computer Science, NASA Ames Research Center, California, March 1986.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

| REPORT DOCUMENTATION PAGE | | READ INSTRUCTIONS BEFORE COMPLETING FORM |
|--|-----------------------|--|
| 1. REPORT NUMBER CSL-TR-87-331 | 2. GOVT ACCESSION NO. | 3. RECIPIENT'S CATALOG NUMBER |
| 4. TITLE (and Subtitle) THROUGHPUT ANALYSIS OF THE IEEE 802.4 TOKEN BUS STANDARD UNDER HEAVY LOAD | | 5. TYPE OF REPORT & PERIOD COVERED Technical Report |
| | | 6. PERFORMING ORG. REPORT NUMBER CSL-TR-87-331 |
| 7. AUTHOR(s) Joseph Pang and Fouad Tobagi | | 8. CONTRACT OR GRANT NUMBER(s) NAG 2-292 NAGW 419 |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS Stanford Electronics Laboratories Stanford University Stanford, CA 94305 | | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS |
| 11. CONTROLLING OFFICE NAME AND ADDRESS National Aeronautics and Space Administration Washington, D.C. 20546 | | 12. REPORT DATE May 1987 |
| | | 13. NUMBER OF PAGES 44 |
| 14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Resident Representative Office of Naval Research DUR 165 Stanford University, Stanford, CA 94305 | | 15. SECURITY CLASS. (of this report) Unclassified |
| | | 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |
| 16. DISTRIBUTION STATEMENT (of this Report) Approved for public releas; distribution unlimited. | | |
| 17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report) | | |
| 18. SUPPLEMENTARY NOTES | | |
| 19. KEY WORDS (Continue on reverse side if necessary and identify by block number) | | |
| 20. ABSTRACT (Continue on reverse side if necessary and identify by block number) It has become clear in the last few years that there is a trend towards integrated digital services. Parallel to the development of public Integrated Services Digital Network (ISDN) is service integration in the local area (e.g. a campus, a building, an aircraft). The types of services to be integrated depend very much on the specific local environment. However, applications tend to generate data traffic belonging to one of two classes. According to IEEE 802.4 terminology, the first major class of traffic is termed synchronous, such as | | |

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-LF-014-6601

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

packetized voice and data generated from other applications with real-time constraints, and the second class is called asynchronous which includes most computer data traffic such as file transfer or facsimile.

In this report, we examine the IEEE 802.4 token bus protocol which has been designed to support both synchronous and asynchronous traffic. The protocol is basically a timer-controlled token bus access scheme. By a suitable choice of the design parameters, it can be shown that access delay is bounded for synchronous traffic. As well, the bandwidth allocated to asynchronous traffic can be controlled. We present a throughput analysis of the protocol under heavy load with constant channel occupation of synchronous traffic and constant token-passing times.